Research report 99

100 EXERCISES IN THE THEORY OF AUTOMATA AND FORMAL LANGUAGES

Wojciech Rytter*

(RR99)

Abstract

We present a collection of a hundred simple problems in the theory of automata and formal languages which could be useful for tutorials and students interested in the subject. Solutions tc these problems require only the knowledge of an introductory course in automata and formal languages which is usually taught for second or third year students of computer science. However some of the exercises require deeper understanding of the subject and some sophistication. Most of the questions are about regular languages and finite automata, and context-free languages and pushdown automata. A small collection of problems concerning various interesting properties of strings is also included in the section 'miscellaneous'. There are no problems related to decidability or the complexity of algorithms. The collection can be useful also because there are presently no exercise-books in the theory of automata and formal languages.

* Institute of Informatics
Warsaw University, and

Department of Computer Science
University of Warwick
Coventry
CV4 7AL, UK

April 1987

100 EXERCISES IN THE THEORY OF AUTOMATA AND FORMAL LANGUAGES

Wojciech Rytter

Institute of Informatics, Warsaw University and
Dept.of Computer Science, University of Warwick

We present a collection of one hundred simple problems in the theory of automata and formal languages which could be useful for tutorials and students interested in the subject . Solutions to these problems require only the knowledge of an introductory course in automata and formal languages which is usually taught for second or third year students of computer science. However some of the exercises require deeper understanding of the subject and some sophistication. Most of the questions are about regular languages and finite automata, and context-free languages and pushdown automata. A small collection of problems concerning various interesting properties of strings is also included in the section 'miscellaneous'. There are no problems related to decidability or to the complexity of algorithms. The collection can be useful also because there are at present no exercise-books in the theory of automata and formal languages.

Many problems were selected from the following books:

S.Ginsburg. The mathematical theory of context-free languages. McGraw-Hill Book Co. (1966)
M.A.Harrison. Introduction to formal languages and automata. Addison-Wesley (1978)
J.E.Hopcroft and J.D.Ullman. Introduction to automata theory, formal languages and computation. Addison-Wesley (1979)
M.Lothaire. Combinatorics on words. Encyclopedia of Mathematics and Its Applications, vol.17, Addison-Wesley (1983)
A.Salomaa. Jewels in formal language theory. Computer Science Press (1981)

Regular languages and finite automata

1. For a given string w of nonzero length n construct an (n+1)-state deterministic finite automaton A_w accepting all strings containing w as a substring. A_w is called the string-matching machine. Take prefixes of w as states of A_w.

2. For a given set of strings $\{w_1,w_2,...w_k\}$ construct a finite automaton A with output such that after reading $a_1....a_n$ A outputs the set of all indices i such that w_i is a suffix of $a_1...a_n$. A should have O(n) states, where $n=|w_1|+... +|w_k|$. We assume that the size of the alphabet and the number k are constants. A is called the pattern-finding machine.
The structure of A can be based on the tree of prefixes of patterns.

3. Let G be a digraph, construct a deterministic finite automaton accepting the set of all paths of G (paths are sequences of nodes, two consecutive nodes have to be adjacent).

4. Construct a deterministic finite automaton accepting the language
$L_q=\{x.y$: x, y are binary strings, $[x.y]_2 \leq q$ },
where q is a given rational numbers and $[x.y]_2$ is a number represented by x.y in binary. The least significant digit is on the right.

5. Prove that if the number q is not rational then the language L_q from the previous exercise is not regular.

6. Construct a deterministic finite automaton accepting the languge

$L=\{xcy$: $(a[x]_2+b[y]_2)$ mod $p = r$, $x,y \in \{0,1\}^+\}$,

where a,b,p,r are given natural numbers .

7. Let U be a regular language, and U(i), i=1..n, be arbitrary languages over the same finite alphabet (U(i) could be nonregular). Construct a deterministic finite automaton accepting the language

$L=\{ i_1i_2...i_k$: $U \supseteq U(i_1)©U(i_2)©...U(i_k)$ }, where © is the operation of language concatenation.

8. Show that if L is accepted by a nondeterministic finite automaton with n states then L is also accepted by a deterministic finite automaton with 2^n states.

9. Show that if L is accepted by an n-state deterministic finite automaton then L^R is accepted by a deterministic finite automaton with at most 2^n states , where $L^R = \{w^R : w \varepsilon L\}$, w^R is the mirror image of the string w, e.g. abc^R=cba.

10. Let $L=\{w \varepsilon \{0,1\}^+ :$ n-th symbol of w is 1 $\}$. Construct an (n+2)-state deterministic finite automaton accepting L, a 2^n-state deterministic finite automaton accepting L^R, and an (n+1)-state nondeterministic finite automaton accepting L^R.

11. Prove that every deterministic finite automaton accepting L^R requires at least 2^n states, where L is the language from the previous exercise.

12. Show that a language accepted by a deterministic finite automaton with n nodes is infinite if and only if it contains a word w of length $n \le |w| < 2n$. Prove also that every regular infinite language contains a language $w_1 w_2^* w_3$ for some strings w_1, w_2, w_3.

13. Prove that for any language L (which may be nonregular) over a finite alphabet the languages sub(L) and sup(L) are regular, where sub(L) is the set of all subsequences of strings in L, and sup(L) is the set of all supersequences of strings in L. For example abc is a subsequence of acbac. (v is a supersequence of w, iff w is a subsequence of v).
Let « denote the relation 'to be a subsequence of', for example acd « badcbbdb. Assume that the alphabet is finite. Use the following fact: every set of pairwise incomparable elements with respect to « is finite; for every finite language L the languages sup(L) and sub(L) are regular.

14. Prove that for any language L (which may be nonregular) over one-element alphabet the language L^* is regular.

15. Let $x \le y$ iff x is a prefix of y. Prove that if L is a regular language then the set max(L) of maximal elements of L (with respect to $\le$) is also a regular language.
Prove that if L is a regular language then the set min(L) of its minimal elements (with respect to $\le$) is also a regular language.

16. Let h be a homomorphism. Prove that the set $\{h(w) : h(w)=h(y)$ for some string y, $w \neq y \}$ is a regular language.

17. We define the star-hight (sh, in short) of regular expressions (with operations of union, concatenation and closure *) as follows. If there is no operation * in the expression E then sh(E)=0, $sh(E^*)=sh(E)+1$, $sh(E1 \cup E2)=\max(sh(E1),sh(E2))$ and $sh(E1 \cdot E2)=\max(sh(E1),sh(E2))$.

For a regular language L we define sh(L) = min{ sh(E) : E is a regular expression describing L }. Prove that for every regular language L over the one-letter alphabet we have $sh(L) \leq 1$.

18. Prove that if L is a regular language then cycle(L) is also regular, where cycle(L)= {uv: vu ε L}.

19. Let L be any regular language whose words have lengths divisible by three. For a string $x=x_1x_2x_3$, where $|x_1|=|x_2|=|x_3|$, denote $x_{(1)}=x_1, x_{(2)}=x_2, x_{(3)}=x_3$.

Let $L_{(i)}=\{x_{(i)}: x \in L\}$, $L_{(i,j)}=\{x_{(i)}x_{(j)}: x \in L\}$.

Prove that $L_{(1)}, L_{(2)}, L_{(3)}, L_{(1,2)}, L_{(2,3)}$ are regular.

20. Prove that there exists a regular language L such that $L_{(1,3)}$ is nonregular.

21. For a given string w of nonzero length n construct a 2n-state deterministic finite automaton accepting the set of all nonempty substrings of w. For a given string w of length n and a string x define S_x to be the set of all positions ending an occurence of x in w. Use the following facts: the sets S_x are nonoverlapping (if $S_x \cap S_y \neq \emptyset$ then S_x is a subset of S_y or S_y is a subset of S_x). There are at most 2n sets S_x.

Use sets S_x as states of an automaton.

22. Prove that if L is regular then the language $\{ x : xx^R \in L \}$ is also regular.

23. Prove that if L is regular then the language $\sqrt{L} = \{x : xx \in L \}$ is regular.

24. Prove that if L is a regular language then the language root(L)=$\{x : x^n \in L$ for some natural $n\}$ is also regular.

25. Prove that if L is regular then $\{x: xy \in L$ for some y with $|y|=|x|^2\}$ is a regular language.

26. Prove that if L is regular then $\{x: xy \in L$ for some y such that $|y|=2^{|x|}\}$ is also a regular language.

27. Let L1 || L2 be the set of all strings w which can be decomposed into two disjoint subsequences w1, w2 such that w1 ε L1 and w2 ε L2, e.g. it can be w=abcabd, w1=cbd and w2=aba. Prove that if L1, L2 are regular then L1 || L2 is also regular.

28. Let $L^{\#} = L \cup L \| L \cup \; L \| L \| L \; \cup$ .... Find a regular language L over a two-letter alphabet such that $L^{\#}$ is nonregular.

29. Prove that every language accepted by a two-way finite automaton is regular. (The two-way finite automaton can move its input head in both directions, left and right.)

30. Construct a finite automaton with output which reading the input $(a_1,b_1)\ (a_2,b_2)\ .....(a_n,b_n)$ outputs $c_1c_2....c_n$, where c=a+b , and a, b, c are numbers whose binary representation are $a_1..a_n$, $b_1...b_n$, $c_1...c_n$, respectively, with a_1,b_1,c_1 being the least significant digits. Assume for simplicity that $a_n=b_n=0$.
Solve the same problem, but this time for c=a-b. Assume that a>b (in the other case assume that the output is meaningless).

31. Prove that the corresponding finite automaton with output does not exist if we replace the operation "+" by multiplication (requiring only n digits of the product to be produced).

32. A set of integers is linear iff it is of the form $\{c+pn : n=0,1,2,... \}$. A set is semilinear iff it is a finite union of semilinear sets. Let R be a regular language. Prove that the set $\{n : a^n \varepsilon R \}$ is a semilinear set.

33. The language L has the finite power property iff $L^k=L^{k+1}$ for some natural n. Prove that a nonempty language L has the finite power property iff $L^k=L^*$ for some natural k.

34. Prove that every infinite regular language L over the one-element alphabet has the finite power property, if ε is in L.

35. If g,h are homomorphisms then denote the language { x : h(x)=g(x) } by E(g,h) . Languages of this type are called equality languages.
Find a nonregular equality language over the alphabet {a,b}.

36. Prove that the language $\{a^ib^j : gcd(i,j)=1 \}$ is not a regular language. (gcd is the abbreviation of 'greatest common divisor')

37. Assume that each production of the grammar G is of the form xA->yB or A->ε, where x,y are strings of terminal symbols, and A,B,C are nonterminal symbols . Prove that the language generated by G is regular.

38. Prove that if A,B are given languages and A does not contain the empty word then the equation $X = AX \cup B$ has only one solution.

39. Let A be an nondeterministic finite automaton. The usual definition of acceptance is the existence of an accepting path. Change this definition as follows: A accepts the input word w iff every possible path of computation ends in an accepting state. Prove that the language accepted by A is regular.

40. Let A be a nondeterministic one-way finite automaton without ε-moves. Assign to the states of A boolean operations 'not', 'and' and 'or'. One can imagine the computation of A for a given input word w as an expression-tree T of possible computation paths, the nodes of T are situations (configurations) which can occur. With each internal node of T associate the same operation as the state of A corresponding to this node, the leaves have boolean values: true, if the corresponding state is accepting, false, otherwise. One can compute now the boolean value of the root of T. We say that A accepts w iff the root of T has the value true. Finite automata with such a definition of acceptance are called alternating finite automata.
Prove that alternating automata accept only regular languages.
Assume that an alternating automaton A has states $s_1,...,s_n$. Let $x=(x_1,...,x_n)$ be vectors of boolean values. Take boolean functions f(x) as states of a constructed automaton. If A is scanning the i-th symbol of the input then f(x) is interpreted as a value of the root of the part of the computation tree of A (of the height i) assuming that the leaf corresponding in this moment to s_i has value x_i, for $i=1..n$.

41. A one-pebble two-way deterministic finite automaton A can move its input head in both directions and has the added capability of marking a tape square (scanned by its input head) by placing a pebble on it. The move of the automaton depends on the present state, scanned input symbol and the presence of the pebble on the tape square scanned. The pebble can be removed and placed later on other square. Prove that two-way deterministic finite automata with one pebble accept only regular languages.
Add two tracks to the input string indicating for each state p, the state q in which A will return if it moves left or right from the given tape cell in state p, under the assumption that the pebble is not encountered. Note that A operating on the augmented tape does not need its pebble. Make use of the fact that if R is a regular language then h(R) is also regular. The homomorphism h can be used to erase the additional tracks.

Context-free languages and pushdown automata

42. Prove that the language $\{a^n b^n c^n : n \geq 1\}$ is not context-free. This is a classical example of a simple language which is not context-free. Prove that every infinite subset of this language is not a context-free language.

43. Construct a context-free grammar generating the complement of the language from the previous exercise.

44. Prove that the language $\{xcy: x,y \in \{a,b\}^+, x=y\}$ is not context-free.

45. Construct a context-free grammar generating the complement of the language from the previous exercise.

46. Prove that the language $\{xx^R : x \in \{a,b\}^+\}$ cannot be accepted by a deterministic pushdown automaton . Let A be a deterministic pushdown automaton. For each string w there is a string w' such that A after reading ww' has the minimal (for all w') height of the stack. This means that A after reading ww' uses only the information (top stack symbol, state) related to ww'. There are two distinct words w_1, w_2 such that this information for w_1w_1' and w_2w_2' is the same. How A reacts on the word $w_1w_1'(w_2w_2')^R$.

47. Prove that the complement of the language from the previous exercise is a context-free language.

48. Construct a one-nonterminal context-free grammar generating the language of all strings over the alphabet $\{a,b\}$ with the same number of a's as b's.

49. Show that the language $\{x\$y$: x is a subword of y; x,y are over the alphabet $\{a,b\}\}$ is not a context-free language.

50. Show that the language $\{x\$y$: x^R is a subword of y; x,y are over the alphabet $\{a,b\}\}$ is context-free.

51. Prove that the complement of the previous language is not context-free.

52. Prove that the language $\{a^i b^j c^k : i \neq j, j \neq k \text{ and } i \neq k\}$ is not a context-free language.

53. Let $L=\{a^n b^k a^n b^k : n,k \geq 1\}$. Is L a context-free language ?

54. Find a context-free language L such that the language $\{x : xy \in L$ for some $y, |x|=|y|\}$ is not context-free.

55. Show that the language $\{x\#y^R : x,y \in \{0,1\}^+, [y]_2=[x]_2+1\}$ is a context-free language.

Is the language $\{x\#y : x,y \in \{0,1\}^+, [y]_2=[x]_2+1\}$ context-free ?

56. We say that a context-free grammar G is self-embedding iff there is a nonterminal symbol A such that $A \to^* xAy$, for some nonempty strings x,y. Prove that if a context-free grammar G is not self-embedding then G generates a regular language.

57. Prove that the set of all possible contents of the pushdown store of a nondeterministic pushdown automaton starting with one-element pushdown store is a regular language.

58. Prove that intersection of a regular language R and a context-free language L is a context-free language.
Prove that if L is deterministic (accepted by a deterministic pushdown automaton) then the resulting language is also deterministic.

59. Find two context-free language's L1,L2 such that L1 || L2 is not a context-free language (see exercise 27 for the definition of the operation ||).

60. Find a context-free language L such that $L^\#$ is not a context-free language (see exercise 28 for the definition of the operation $^\#$).

61. Prove that if X, Y are regular languages then the language $\bigcup_{n\geq 1} (X^n \cap Y^n)$ is a context-free language.

62. Find regular languages X, Y such that $\bigcup_{n\geq 1} (X^n \cap Y^n) = \{a^n b^n : n\geq 1\}$.

63. Find regular languages X, Y and Z such that the language $\bigcup_{n\geq 1} (X^n \cap Y^n \cap Z^n)$ is not context-free.

64. Find a context-free language L such that the language $\sqrt{L}$ is not context-free ($\sqrt{}$ is the operation from exercise 23).

65. Let $h(a)=a$, $h(b)=b$, $g(a')=a$, $g(b')=b$ and $h(x)=g(x)=\varepsilon$ for all other symbols x. The language $L = E(g,h)$ is over the alphabet $\{a,b,a',b'\}$ and it is called the twin shuffle language. Every word in

L is a shuffle of two copies of the same word. Prove that the language L is not context-free.

66. Prove that if U is regular then $\{xy^R: x \neq y,\ xy \in U\}$ is a context-free language.

67. Prove that if L is a context-free language and R is a regular language then $L \parallel R$ is a context-free language (see exercise 27 for the definition of the operation $\parallel$).

68. Prove that every context-free language over the one-element alphabet is regular.

69. A language has the prefix property iff for each two strings one is a prefix of another. Prove that every context-free language with the prefix property is regular.

70. Prove that if L is a context-free language then $\{a^{|w|} : w \in L\}$ is a regular language.

71. Let $\{a,b\}^* \supseteq L$ be a regular language, and h_1, h_2 be homomorphisms. Prove that

$\{h_1(u)c(h_2(u))^R : u \in U\}$ is a linear context-free language.

The language is linear iff it can be generated by a context-free grammar in which right sides of productions contain at most one nonterminal.

72. Prove that if L is a linear language and R is a regular language then $L \cap R$ is a linear language.

73. Prove that the language from exercise 48 is not a linear language.
Use a stronger version of the 'uvwxy' lemma (pumping lemma), where $|uvxy|$ is bounded by a constant.

74. Let h1, h2 be two homomorphisms whose values are words not containing the symbol '\$'. Prove that the languages $\{x\$y^R: h(x)=h(y)\}$ and $\{x\$y^R: h(x) \neq h(y)\}$ are linear context-free languages.

75. Prove that if L is a deterministic context-free language then the language min(L) is also a deterministic context-free language (the operation min is the same as in exercise 16).

76. Let $L=\{a^ib^jc^k : k \geq i \text{ or } k \geq j\}$. Show that the set min(L) is not a context-free language. Consequently L is not a deterministic context-free language.

77. Find a context-free language L such that max(L) is not a context-free language. Use a language similar to the one from exercise 76.

78. Let the language $L \subsetneq \{xcy : x,y \in \{a,b\}^*\}$ be generated by a linear context-free grammar, such that the only production without nonterminal on the right side is of the form A -> c. Prove that $\{x : xcy \in L$, for some $y\}$ is a regular language.

79. Prove that if $\{a,b\}^* \supseteq U$ and the language $L=\{b^{|x|}cx : x \in U\}$ is a context-free language then U is regular. Let A be a pushdown automaton accepting L. Prove that A can be simulated by a one-turn pushdown automaton A' (whose stack changes its mode from nondecreasing to nonincreasing only when the symbol c is read). Then simulate by a finite automaton a part of the computation of A' on the suffix cx of the input.
If we remove the letter c then U can be nonregular, consider $L=\{b^{3n}a^n : n \geq 1\}$.

80. Prove that if L is a context-free language then cycle(L) is also a context-free language (the operation cycle was defined in exercise 18).

81. For given homomorphisms h and g describe context-free languages L1, L2 and a homomorphism f such that $E(h,g)=f(L1 \cap L2)$. Hence $E(h,g)=\emptyset$ if and only if $L1 \cap L2 = \emptyset$. The problem of checking emptiness of the language E(h,g) is called the Post correspondence problem.

82. A two-way deterministic pushdown automaton A (2dpda A, in short) is a deterministic pushdown automaton whose input head can move in two-directions (left and right) and can detect the left and right end of the input word. Construct a 2dpda which accepts the language $\{x\$y : x$ is a subword of $y\}$ (the string matching problem)

83. Let $P=\{xx^R : |x| \in \{a,b\}^+\}$ and V be the set of all words over the alphabet $\{a,b\}$. P is the set of nontrivial palindroms of even length over the alphabet $\{a,b\}$.
Construct a 2dpda A which accepts the language PV (the language of prefix palindroms).

84. Construct a 2dpda which accepts the language P^3V. Use the following fact:

if $w \in P^*$, first(w) = min{ |x| : w=xy, $x \in P$} and parse(w)= min{ |x| : w=xy, $x \in P$, $y \in P^*$} then first(w)=parse(w).
A is recomputing many times end-positions of the first and second prefix palindrom.

85. Construct a 2dpda which accepts the language P^2. Use the following fact: if $x \in P^2$ then x=yz for some y,z such that $y,z \in P$ and y is the longest prefix of x which belongs to P, or z is the longest suffix of x which belongs to P.
Use the 2dpda from exercise 83 to compute the longest prefix palindom and the longest suffix

palindrom.

86. Construct a 2dpda which accepts the language $\{a^n b^{p(n)} : n \geq 1\}$, where p is a given polynomial with natural coefficients.

Miscellaneous

87. The period of the string u is the smallest word v such that u is a prefix of v^k for some k. Prove that if x has periods of sizes p, q and $p+q \leq |x|$ then x has a period of size gcd(p,q).
Use Euclid algorithm.
We say that v is a full period of u iff $u=v^k$ for some k. Prove that xy=yx if and only if x and y have the same smallest full period.Let gcd'(u,v) denote the smallest common full period of u,v, if it exists. Show that gcd'(u,v) exists iff uv=vu. If it exists then $|gcd'(u,v)|=gcd(|u|,|v|)$.

88. Let $f_1=b$, $f_2=a$ and $f_{n+2}=f_{n+1}f_n$. The strings f_n are called Fibonacci words. Prove that $c(f_{n-1}f_n)=f_nf_{n-1}$, where c is the operation of interchanging the last two letters of the word.

89. Let $P_1=\{x : x = x^R, |x|>1, x \in \{a,b\}\}$. P_1 is the set of nontrivial palindroms. The elements of $P_1{}^+$ are called palstars. For every palstar w define

$\mathrm{first}_1(w) = \min\{ |x| : w=xy, x \in P_1\}$ and $\mathrm{parse}_1(w)= \min\{ |x| : w=xy, x \in P_1, y \in P_1{}^*\}$.

Prove that $\mathrm{first}_1(w) \in \{\mathrm{parse}_1(w), 2\,\mathrm{parse}_1(w)+1, 2\,\mathrm{parse}_1(w)-1\}$.

90. Let h be a homomorphism whose domain is $\{a,b\}^*$. Prove that h is one-to-one if and only if $h(ab) \neq h(ba)$.

91. Let h(a)=ab,h(b)=ba. Prove that $h^i(a)$ is a prefix of $h^{i+1}(a)$. Define H to be the infinite word $a_0a_1a_2....$ such that each $h^i(a)$ is its prefix. Prove that the n-th symbol of H is a if and only if the number of ones in the binary representation of n is even. Prove that $h^{i+1}(a)=h^i(a)\,Q(h^i(a))$, where Q is a homomorphism Q(a)=b, Q(b)=a.

92. What is the cardinality of the set of all square-free (not containing subword of the form xx) words over the two-letter alphabet .

93. Let $\{ab,ba\}^* \supseteq L$. Prove that if $x \in L$ then axa and bxb do not belong to L.

Using this result prove that if w does not contain a subword of the form cvcvc ($c \in \{a,b\}$), then h(w) has the same property, where h is the homomorphism from problem 91.

94. Using the result of the previous exercise prove that the infinite word H contains no subword of the form cvcvc ($c \in \{a,b\}$). Hence H is cube-free (has no subword of the form xxx).

95. Let $\partial(a)=a, \partial(b)=ab, \partial(c)=abb$. Using the result of the previous problem prove that the infinite word $M=\partial^{-1}(H)$ is well defined and M is square-free. Conclude that the set of square-free words over three-letter alphabet is infinite.

96. The language L is a code if any product of words from L can be "decoded" in a unique way: if the words w_i, v_i are from L and $w_1w_2...w_k=v_1v_2...v_j$ then $k=j$, and $v_i=w_i$, for $i=1..k$.
Let code-indicator of a word w be $k^{-|w|}$, where k is the size of the alphabet. Let code-indicator (ci(L), in short) of the language L be the sum of code-indicators of words in L.
Prove that if L is a code then $ci(L) \leq 1$.

97. The language L is said to be commutative iff xy=yx for every two words from L. Prove that if L is commutative then $w^* \supseteq L$, for some string w.

98. Let R be a given symmetric binary relation on the alphabet. We write $x \approx y$ iff the string x can be obtained from y by applying several times the operation of exchanging adjacent letters a,b such that R(a,b) holds. Let $h_{a,b}$ be a homomorphism ereasing all symbols except a,b. Prove that
$x \approx y$ if and only if the following two conditions are satisfied:
(i) for each letter a $card_a(x)=card_a(y)$;
(ii) for each pair of symbols a,b if R(a,b) does not hold then $h_{a,b}(x)=h_{a,b}(y)$.
($card_c(x)$ denotes the number of occurences of the letter c in x).

99. Let $X \cap Y=\emptyset$. Let R(a,b) hold iff ($a \in X$, $b \in Y$) or ($a \in Y$, $b \in X$). Prove that if L is a regular language over $X \cup Y$ then $\{uv : u \in X^*, v \in Y^*, uv^R \approx w \text{ for some } w \in L\}$ is a context-free language.

100. The language L is said to be bounded iff L is a subset of $w_1^*w_2^*...w_k^*$ for some words $w_1,...,w_k$. Prove that the language $\{a,b\}^*$ of all words over the alphabet {a,b} is not bounded. Consider the words $w_j = aba^2b^2a^3b^3...a^jb^j$, for $j \geq 1$.